Computational Aspects of Simulating a Wind Turbine Blade Deflection Sensing System

Pedersen, Gert F.; Franek, Ondrej

Published in:
2018 International Workshop on Computing, Electromagnetics, and Machine Intelligence (CEMi'18)

DOI (link to publication from Publisher):
10.1109/CEMI.2018.8610574

Publication date:
2019

Document Version
Accepted author manuscript, peer reviewed version

Computational Aspects of Simulating a Wind Turbine Blade Deflection Sensing System

Gert Frølund Pedersen[1] and Ondřej Franek[1]

[1]*APMS Section, Department of Electronic Systems, Aalborg University, DK-9220 Aalborg Øst, Denmark*
gfp@es.aau.dk

***Abstract*— Recently introduced wind turbine blade deflection sensing system using ultrawideband technology is described. Challenges of simulating microwave propagation along the blade using the finite-difference time-domain method are outlined. Two straightforward techniques that have been used to speed up the FDTD simulation are presented: load balancing of parallel processes and the moving frame technique. In addition, a simple method to reduce the numerical dispersion error of the FDTD method is presented.**

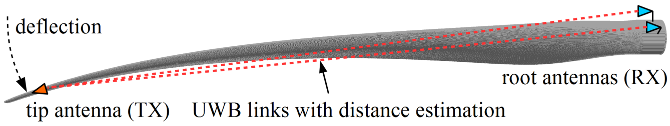

Fig. 1. Principle of the wind turbine blade deflection sensing system with distance estimation using time of arrival on UWB wireless links.

I. Introduction

Wind energy is one of the solutions to reducing our CO_2 footprint, but the cost of generating electricity from wind is still relatively high. One way how to reduce the cost is to install larger rotors with longer blades, but this comes with increased risk of tower strike due to the blade flexibility, and therefore the turbine is not allowed to operate in the strongest winds when most of the energy could be harvested. Our proposed solution is a *blade deflection sensing system*, which will provide early warning for possible tower strike and consequently allow the turbine into higher loads.

The deflection sensing system [1] consists of two antennas at the tip of the blade and two antennas at its root, and a cable inside the blade that feeds the tip antennas. Bending of the blade is determined by triangulation based on the time of arrival of an ultrawideband (UWB) pulse covering the 3–5 GHz band launched by the tip antennas and received at the root antennas (see Fig. 1). Unfortunately, the tip antenna must be placed inside the blade due to aerodynamic noise and exposure to lighting, and the pulsed wave needs to traverse the fiberglass shell at very high angle of incidence and then propagate almost parallel to the blade surface. As a result, there will be many multipath components in the received waveform and the design of the wireless link becomes extremely challenging.

In order to predict the link budget, we have simulated the propagation in our in-house finite-difference time-domain (FDTD) code. We have shown that despite the challenges the FDTD method is capable of delivering results that are in good agreement with measurements [2]. However, due to the enormous size of the blade (58.7 m) and necessary spatial resolution (5 mm), each simulation of the blade took 17 hours to finish. Moreover, the numerical dispersion of the Yee-FDTD scheme [3] caused substantial distortion of the received pulse. In this paper, we present the measures that have been taken to reduce the computation time of the simulations and to minimize the error due to numerical dispersion.

II. Load Balancing

The simulation of the blade required using the perfectly matched layers (PML) as an outer boundary imitating open space. The PML thickness was 50 cells due to the elongated shape of the computational domain and grazing incidence of the propagating waves. Since our in-house FDTD code uses MPI library for deployment on parallel clusters, the domain must be partitioned into regular segments. However, the corner segments contained large portions of the PML layers where more computations need to be performed per FDTD cycle. The entire simulation was then slowed down because other segments needed to wait in each time step for the corner "heavy" ones.

The solution to this problem was to perform load balancing of the segments [4] using a very simple algorithm. The benefit was 22 % shorter running time of the simulation, by changing only the boundaries of the MPI segments in the code, without modifying the algorithm of the method itself.

III. Moving Frame FDTD Method

Running times have been significantly reduced by implementing the moving frame FDTD technique [5]. Since all important information of the short UWB pulse is contained in a very small volume of space at each time instant (see Fig. 2), it is possible to reduce the computational burden by calculating the FDTD update equations only in a subdomain (frame) that is moving along the length of the blade as time progresses. Fields ahead of the frame do not have to be calculated because the pulse did not reach there yet. On the other hand, fields behind the frame need not be calculated either because all scattering from this area arrives after the important part of the pulse.

The z-coordinate frame boundaries are calculated at each time instant t in the simulation by

$$z_1 = z_{\mathrm{TX}} - v_{\mathrm{g}} t \quad (1)$$

$$z_2 = z_1 + v_{\mathrm{g}} T + \mathrm{buffer} \quad (2)$$

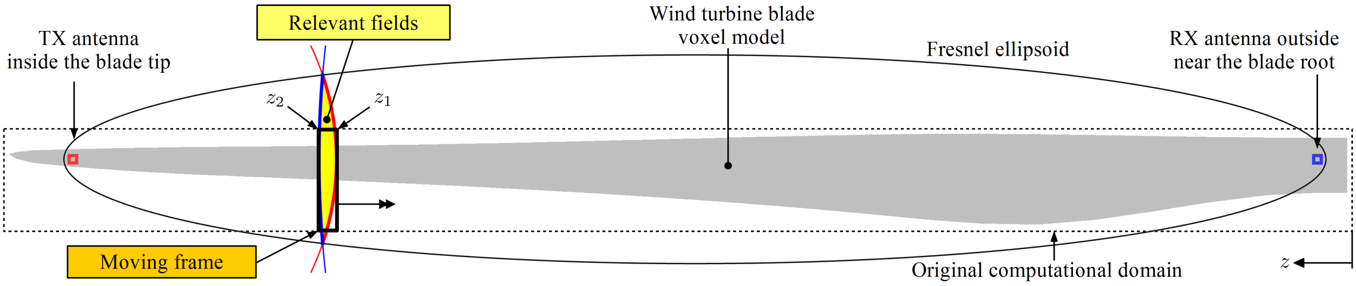

Fig. 2. Principle of the Moving Frame FDTD method. The relevant fields (yellow) are enclosed between the pulse front (red) and pulse back (blue) spheres. As a result, only the fields inside the moving frame (bold outline) need to be calculated.

where v_g is the group velocity of propagation in the z-direction along the FDTD grid (lower than speed of light due to numerical dispersion), T is the time interval of interest in the output pulse, and z_{TX} is the z-coordinate of the TX antenna. The buffer accounts for slower propagation when the pulse traverses the fiberglass shell of the blade.

The pulse obtained using the moving frame technique is in excellent agreement with FDTD applied on the full domain, see Fig. 3. The running times have been reduced from the original 17 hours to 1.5 hours on our parallel cluster with 288 cores, which allows for much faster optimizations.

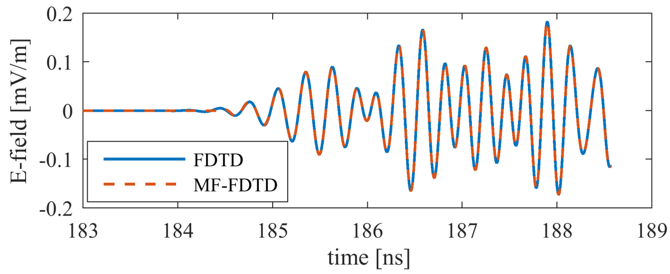

Fig. 3. Comparison between fields obtained by the original full domain FDTD (blue) and the Moving Frame FDTD (red dashed) methods.

IV. Dispersion Compensation

The pulse waveform as seen in Fig. 3 is affected by numerical dispersion inherent in the FDTD method. To obtain more accurate results, a dispersion compensation [6] can be applied to the output waveform E

$$E^{(\mathrm{comp})} = E \cdot \mathrm{e}^{\mathrm{j}(\tilde{k}_z - k)r} \tag{3}$$

where r is the distance between TX and RX, $k = \omega/c$ is the free space (ideal) wavenumber and

$$\tilde{k}_z = \frac{2}{\Delta z} \arcsin\left(\frac{\Delta z}{c\Delta t} \sin\frac{\omega\Delta t}{2}\right) \tag{4}$$

is the dispersive wavenumber due to FDTD propagation along the z-coordinate of the grid. Comparison of the dispersion compensated waveform with measured pulse on a real blade is shown in Fig. 4.

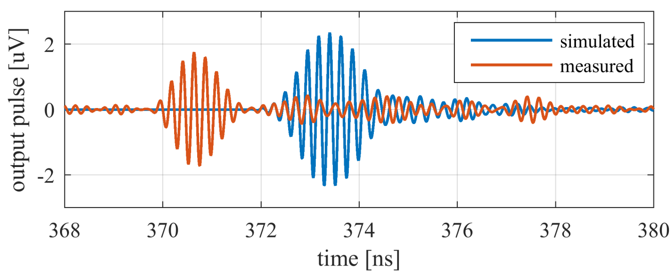

Fig. 4. Comparison between output signals simulated with dispersion compensation (blue) and measured (red). The delay includes also the traveling time in the cable.

V. Conclusions

FDTD Simulation of UWB propagation along large irregularly shaped dielectric bodies, such as a wind turbine blade, can be challenging in terms of computational resources and accuracy. However, as we have shown, even by applying relatively simple techniques and modifications, the running times can be reduced, sometimes quite dramatically, and more accurate results can be obtained by compensating for the numerical dispersion error.

Acknowledgment

This work was supported by the Innovation Fund Denmark (“iRotor” project) and the Danish e-Infrastructure Cooperation (DeIC).

References

[1] S. Zhang, T. L. Jensen, O. Franek, P. C. Eggers, K. Olesen, C. Byskov, and G. F. Pedersen, “UWB wind turbine blade deflection sensing for wind energy cost reduction,” *Sensors*, vol. 15, no. 8, pp. 19 768–19 782, 2015.

[2] O. Franek, S. Zhang, K. Olesen, P. C. Eggers, C. Byskov, and G. F. Pedersen, “Numerical modeling of ultrawideband propagation along a wind turbine blade,” *IEEE Transactions on Antennas and Propagation*, In press.

[3] A. Taflove and S. C. Hagness, *Computational electrodynamics: The Finite-Difference Time-Domain Method*, 3rd ed. Boston: Artech house, 2005.

[4] O. Franek, “A simple method for static load balancing of parallel FDTD codes,” in *2016 International Conference on Electromagnetics in Advanced Applications (ICEAA)*. IEEE, 2016, pp. 587–590.

[5] ——, “Numerical modeling of a wind turbine blade deflection sensing system using the moving frame FDTD method,” in *2018 International Conference on Electromagnetics in Advanced Applications (ICEAA)*. IEEE, 2018.

[6] O. Franek, S. Zhang, K. Olesen, P. C. F. Eggers, C. Byskov, and G. F. Pedersen, “Application of numerical dispersion compensation of the Yee-FDTD algorithm on elongated domains,” in *2017 International Conference on Electromagnetics in Advanced Applications (ICEAA)*. IEEE, 2017, pp. 1243–1246.